FIG. **9** shows another example of the recursive dilation method of this invention for an exemplary 8×8 boundary block, where a horizontal region boundary is located between rows **1** and **2**. The data above the region boundary in FIG. **8** is the missing data. According to the methods and systems of this invention, the recursive dilation method assigns, for each column m, the value of the pixel (m,n+1), which is the picture pixel in the picture portion adjacent to the boundary, to the missing value at each pixel (m,n) for n<2.

FIG. **10** shows another example of the recursive dilation method of this invention for an exemplary 8×8 boundary block, where a vertical region boundary is located between columns **4** and **5**, and where a horizontal region boundary is located between rows **5** and **6**. The data to the right of the vertical region boundary and below the horizontal region boundary is the missing data. It should be appreciated that if the non-missing data that is both above and to the left of the region boundaries is a first data type, either or both of the vertical and horizontal region boundaries could be between different types of data. That is, the missing data to the right of the vertical boundary could be a second data type, while the missing data below the horizontal boundary could be a third type of data, and the missing data that is both below and to the right of the horizontal and vertical boundaries could be a fourth type of data.

According to the methods and systems of this invention, the recursive dilation method assigns, for each column m and each row n, the value of the pixel (m,n−1), which is the picture pixel in the picture portion adjacent the horizontal region boundary, to the missing value at each pixel (m,n) for n>5 and m<5, while the value of the pixel (m−1,n) in the picture portion adjacent the vertical region boundary is assigned to the missing value at each pixel (m,n) for m>4 and n<6. The missing value at pixel (m,n) for m>4 and n>5 may be assigned the value of its neighbor that is the value of closest boundary pixel. In this instance, the values of both the left neighbor and the upper neighbor equal y45.

FIG. **11** shows another example of the recursive dilation method of this invention for an exemplary 8×8 boundary block, where a vertical region boundary is located between columns **4** and **5**, and where a horizontal region boundary is located between rows **2** and **3**. The data to the right of the vertical region boundary and below the horizontal region boundary is the missing data. According to the methods and systems of this invention, the recursive dilation method assigns for each column m and each row n, the value of the each pixel (m−1,n), which is the pixel in the picture portion adjacent the vertical region boundary, to the missing value at pixel (m,n) for m>4, while the value of each pixel (m,n−1) in the picture portion adjacent the horizontal boundary is assigned to the missing value at each pixel (m,n) for n>2. The missing values at the pixel (5,3) may be assigned either the value of the left neighbor, the pixel (4,3) or its upper neighbor, the pixel (5,2). If both of the left neighbor and the upper neighbor are missing data, then the assigned values of the closest boundary pixel may be given to the pixel. Thus, for m>5 and n>3, each pixel (m,n) may assigned a value of the closest of the pixel (m−1,n) or the pixel (m,n−1). If both the pixel (m−1,n) and the pixel (m,n−1) are of the same proximity, the value of either may be applied. Thus, the missing values at the pixel (**6**,**4**) may be assigned the value of the pixel (**4**,**4**) or the pixel (**6**,**2**). Similarly, the missing value of the pixel (**7**,**5**) may be assigned the value of the pixel (**4**,**5**) or the pixel (**7**,**2**).

In the examples described above, the pixels containing missing data are assigned the values of the boundary pixels closest to the pixel containing the missing data. It is to be appreciated that any conceivable method may be applied in which the values of the closest boundary pixels are assigned to the pixel containing the missing data. In accordance with the methods and systems of this invention, the observable artifacts caused by the missing data is eliminated.

As shown in FIG. **2**, the encoder **400** is preferably implemented on a programmed general purpose computer. However, the encoder **400** can also be implemented on a special purpose computer, a programmed microprocessor or microcontroller and peripheral integrated circuit elements, an ASIC or other integrated circuit, a digital signal processor, a hardwired electronic or logic circuit such as a discrete element circuit, a programmable logic device such as a PLD, PLA, FPGA or PAL, or the like. In general, any device, which is capable of implementing steps **S1300**–**S1400** of FIGS. **5** and **6**, can be used to implement the encoder **400**.

Similarly, though FIG. **2** shows the decoder **500** being preferably implemented on a programmed general purpose computer, the decoder **500** can also be implemented on a special purpose computer, a programmed microprocessor or microcontroller and peripheral integrated circuit elements, an ASIC or other integrated circuit, a digital signal processor, a hardwired electronic or logic circuit such as a discrete element circuit, a programmable logic device such as a PLD, PLA, FPGA or PAL, or the like. In general, any device, which is capable of implementing steps **S1600**–**S1700** of FIGS. **5** and **7**, can be used to implement the decoder **500**.

This invention has been described in connection with the preferred embodiments. However it should be understood that there is no intent to limit the invention to the embodiments described above. On the contrary, the intent to cover all alternatives, modification, and equivalents as may be included within the spirit and scope of the invention.

What is claimed is:

1. A method for compressing an image, comprising:

dividing the image into image segments, the image segments including at least one segment of a first type and at least one segment of a second type, wherein at least each at least one segment of the first type is divided into image blocks and compressed on a block-by-block basis, and at least one of the image blocks of the at least one segment of the first type contains a boundary between data of the first type and missing data, each image block comprising a plurality of pixels, and each pixel having a value;

identifying each image block containing a boundary with missing data;

recursively replacing the missing data of each first type boundary block with first type data by assigning the value of a closest boundary pixel;

compressing each first type image block and each first type boundary block using a first compression technique; and

compressing each at least one segment of the second type using a second compression technique.

2. The method of claim **1**, wherein the step of recursively replacing comprises determining, for each pixel of the missing data of the first type boundary block, whether that pixel is adjacent to a boundary pixel of the first type.

3. The method of claim **2**, wherein assigning the value of the closest boundary pixel comprises assigning the value of the boundary pixel to the pixels of the missing data of the first type boundary block determined to be adjacent to the boundary pixel.